REMARKS

Reconsideration and allowance are requested.

Claims 43-62 stand rejected under 35 U.S.C. §101 as allegedly being directed to non-statutory subject matter. This rejection is respectfully traversed.

Applicants adopt the Examiner's suggested change to claim 43. The rejection under 35 U.S.C. §101 should now be moot.

Claims 1-17, 21-38, 42-58, and 62 stand rejected for obviousness under 35 U.S.C. §103 based on USP 5,598,544 to Ohshima in view of USP 5,802,598 to Watt. This rejection is respectfully traversed.

The technology of this application is directed to data processing systems that can execute a variable length instruction stored in distinct memory locations. The first part of a variable length instruction is stored in a first memory location, and the second part of the instruction is stored in a second different memory location. A "fix-up" memory address region is used to bring together the two parts of the single instruction. When a two-part, variable-length instruction occurs, the program execution flow is temporarily diverted to the fix-up memory to read the freshly reconstructed variable length instruction stored there. Thereafter, the program execution flow is returned to reading from the normal memory.

Ohshima is concerned with a very different problem: efficient processing of an instruction that includes both basic and expanded segments. A basic segment contains a code indicating the type of instruction (basic or expanded), and an expanded segment contains information relevant to the type of instruction specified by the basic segment. One instruction is formed from one or more basic and expanded segments. See column 3, lines 31-40. Even though the basic segments have a known length, the length of expanded segments can vary, and

this length is not known until its corresponding basic segment has been decoded. Hence, in a situation where one of Ohshima's instructions consists of two basic segments and the first basic segment is followed by an expanded segment, the second basic segment cannot be input into a decoder until after the first basic segment has been decoded, which allows the length of the expanded segment to be determined. Thus, two cycles are required to decode the single instruction. See column 2, line 59 to column 3, line 9.

Instead of repeating them here, the distinctions set forth in Applicants' Appeal Brief are incorporated into this response by reference. The following remarks focus on the new constructions/interpretations adopted by the Examiner in this most recent office action.

The Examiner appears to be mapping the claimed memory and the claimed various memory address regions to the instruction buffer 2 of Ohshima and the plurality of entries 404, 404', 404'' etc. within the instruction buffer 2. But the memory recited in the independent claims is the memory in which the variable length instructions are stored prior to any fetching as part of decoding and execution. In this regard, the independent claims are amended to recite: "executing a sequence of variable length instructions stored prior to any fetching as part of decoding and execution within a plurality of discrete memory address regions within a memory of a data processing apparatus." Example support for this amendment is found in Figure 2 in which fetching from the memory 54 occurs before the instructions are written into the decoder buffer 64 and passed to the Javacard decoder 58 and the execution engine 56 of Figure 5. In contrast, Ohshima's instruction buffer 2 does not store the variable length instructions prior to any fetching as part of decoding an execution. Rather, storing the variable length instructions in the instruction buffer 2 is an integral part of decoding and execution in Ohshima's system.

Regarding the plurality of discrete memory address regions, the Examiner maps them onto Figure 7A of Ohshima which shows a plurality of contiguous entries which do not have a gap therebetween. The independent claims are amended to clarify that the claimed discrete memory address regions are discrete in the sense of being non-contiguous with a gap therebetween. Example support is found in the illustration in Figure 1 of the application and the description on page 3, lines 26 to 28. The adjacent and contiguous entries 4 and 4' in Figure 7A of Ohshima are not a "plurality of discrete memory address regions including a current memory address region and a following memory address region, said current memory address region and said following memory address region <u>being non-contiguous with a gap therebetween</u>."

The independent claims are further amended to emphasize yet another distinction from Ohshima relating to the fix-up memory address region and <u>copying</u> (concatenating is replaced by copying of the different portions of the instruction data into this fix-up memory address region. The fix-up memory address region is specified as "being separate from said current memory address region and said following memory address region within said memory." An example of this is illustrated in Figure 4 of the instant application.

The claimed fix-up memory is not taught by operation of the read out position pointer P identified in the office action and referred to in column 2, lines 38 to 50 of Ohshima and illustrated in Figure 5. Merely manipulating a read out pointer does not involve copying; nor does it provide a fix-up memory address region that is separate from the current memory address region and the following memory address region within the memory.

With regard to the combination of Ohshima and Watt, the Examiner's argument is difficult to follow. The independent claims recite (1) detecting an attempt to execute a variable length instruction spanning two discrete memory address regions where a variable length

instruction spans two discrete memory address regions and (2) triggerring a memory abort. When both these conditions (1) and (2) are detected, then the claimed further steps/operations are taken. The claims are amended to emphasize this conditional relationship between steps/operations.

The Examiner's argument appears to be that detecting memory aborts is generally a good idea and accordingly the person of ordinary skill in the art might consider combining Ohshima and Watt. But this general idea falls short of teaching that the concatenation and the use of the fix-up memory should occur in circumstances when a memory abort has been triggered due to the variable length instruction spanning two discrete and non-contiguous memory address regions with a gap therebetween. The operation of Ohshima which the Examiner attempts to compare with the claimed concatenation has nothing to do with the generation of memory aborts or the separation of discrete memory address regions. Simply applying memory protection with memory aborts to Ohshima's system would not result in these memory aborts being a precondition for the claimed concatenation and the use of the claimed fix-up memory.

The application is in condition for allowance. An early notice to that effect is requested.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: _____
John R. Lastova
Reg. No. 33,149

JRL:maa
901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100